What is claimed is:

1. A method for identifying data processing systems within a network having a vulnerability, comprising the steps of:

computing a set of hash values derived from and representing a set of resources distributed across a plurality of data processing systems within a network;

storing, at a first data processing system within the network, the computed set of hash values together with an identification of the respective one of said plurality of data processing systems storing a resource corresponding to each computed hash value;

in response to an indication that a first resource is associated with a specific vulnerability, comparing at least one hash value representing the first resource with the stored set of hash values to identify matching hash values, and using the identification of matching hash values and the stored identification of respective systems to determine the systems within the plurality of data processing systems storing replicas of the first resource.

2. The method of claim 1, wherein the first resource is a collection of component resources and said at least one hash value comprises a logical combination of hash values representing each of the component resources.

3. The method of claim 1 wherein the vulnerability is a vulnerability to a computer virus.

4. The method of claim 2 wherein the vulnerability is a vulnerability to computer hacking.

5. The method of claim 1, further comprising classifying the systems storing replicas of the first resource as vulnerable.

6. The method of claim 1, further comprising:

replacing the replica of the first resource at each of the systems determined to be storing a replica of the first resource.

7. The method of claim 1, further comprising:

patching the replica of the first resource at each of the systems determined to be storing a replica of the first resource.

8. The method of claim 7, further comprising:

prior to patching the replica of the first resource with patch code, comparing a set of hash values representing all pre-requisite programs of the patch code with the stored set of hash values to identify matching hash codes; and

in response to identification of matching hash codes for all pre-requisite programs, determining that said patching of the replica of the first resource with the patch code should proceed.

9. The method of claim 1, further comprising:

sending a notification of the vulnerability to each system determined to be storing a replica of the first resource.

10. The method of claim 9, further comprising:

responding to the determination of respective systems storing replicas of the first resource by selecting a sequence of vulnerability-resolution instructions relevant to the vulnerability and including the selected instructions within the notification sent to each data processing system.

11. The method of claim 1, including the step of receiving, from a remote data processing system, at least one hash value representing a first resource associated with a vulnerability together with vulnerability resolution information.

12. The method of claim 11, wherein the vulnerability resolution information comprises at least one program code patch for removing the vulnerability.

13. The method of claim 1, including the step of computing the at least one hash value representing the first resource in response to said indication that the first resource is associated with the vulnerability.

14. The method of claim 1, including the step of receiving the at least one hash value at the first data processing system together with the indication that the first resource is associated with the vulnerability.

15. A data processing apparatus comprising:

a data processing unit;

a data storage unit;

a repository manager configured to store a set of hash values and associated system identifiers in a repository within the data storage unit, wherein the set of hash values are derived from and represent a set of resources distributed across a plurality of data processing systems and the system identifiers identify particular systems within said plurality of data processing systems at which the resources are stored; and

a vulnerability coordinator configured to respond to an indication that a first resource has a vulnerability, by comparing at least one hash value representing the first resource with the stored set of hash values to identify matching hash values, and configured to use the identification of matching hash values and stored system identifiers to identify systems within the plurality of data processing systems storing replicas of the first resource.

16. The data processing apparatus of claim 15, wherein the vulnerability coordinator is configured to receive at least one hash value representing a first resource from a second data processing apparatus.

17. The data processing apparatus of claim 15, wherein the vulnerability coordinator is configured to compute at least one hash value representing the first resource.

18. A distributed data processing system comprising:

a plurality of client data processing systems each comprising a data processing unit and a data storage unit storing resources; and

a server data processing system comprising a data processing unit; a data storage unit; a repository manager configured to store a set of hash values and associated system identifiers in a repository within the data storage unit, wherein the set of hash values are derived from and represent a set of resources distributed across the plurality of client data processing systems, and the system identifiers identify particular systems within said plurality of data processing systems at which the resources are stored; and a vulnerability coordinator which is configured to respond to an indication that a first resource has a vulnerability, by comparing at least one hash value representing the first resource with the stored set of hash values to identify matching hash values, and to use the identification of matching hash values and stored system identifiers to identify systems within the plurality of data processing systems storing replicas of the first resource.

19. A computer program product, comprising program code recorded on a recording medium,

for controlling the performance of operations on a data processing system on which the program code executes, the program code comprising:

a repository manager configured to store a set of hash values and associated system identifiers in a repository, wherein the set of hash values are derived from and represent a set of resources distributed across a plurality of data processing systems and the system identifiers identify particular systems within said plurality of data processing systems at which the resources are stored; and

a vulnerability coordinator configured to respond to an indication that a first resource has a vulnerability, by comparing at least one hash value representing the first resource with the stored set of hash values to identify matching hash values, and to use the identification of matching hash values and stored system identifiers to identify systems within the plurality of data processing systems storing replicas of the first resource.

20. A method for determining whether a data processing system has a vulnerability, comprising the steps of:

computing a set of hash values representing a set of resources of the data processing system;

for a resource associated with the vulnerability, comparing at least one hash value representing the vulnerability-associated resource with the computed set of hash values, to identify matching hash values; and

determining, from said identification of matching hash values, whether the data processing system includes the resource associated with the vulnerability; and

in response to determining that the data processing system includes the resource associated with the vulnerability, classifying the data processing system as vulnerable.

21. The method of claim 20, wherein the data processing system is a first data processing system connectable to a second data processing system via a network, and the method further comprises:

in response to determining that the first data processing system includes the resource associated with the vulnerability, retrieving vulnerability-resolution instructions relevant to the vulnerability from the second data processing system.

22. The method of claim 21, further comprising:

executing the vulnerability-resolution instructions on the first data processing system.

23. The method of claim 20, wherein the data processing system is a first data processing system connectable to a second data processing system via a network, and the method further comprises:

in response to determining that the first data processing system includes the resource associated with the vulnerability, retrieving patching code relevant to the vulnerability from the second data processing system.

24. The method of claim 23, further comprising:

executing the patching code on the first data processing system.

25. The method of claim 20, further comprising:

reporting the vulnerability to a vulnerability resolution manager.

26. A computer program product comprising program code recorded on a recording medium for controlling operations within a data processing apparatus, wherein the program code comprises:

a hashing function for generating hash values representing data processing system resources; and

a vulnerability determination program configured to compare at least one hash value representing a resource associated with a vulnerability with a set of hash values representing resources of the data processing apparatus, thereby to identify matching hash values, and configured to use the identification of matching hash values to determine whether the data processing apparatus includes the resource associated with the vulnerability.

27. The computer program product of claim 26, wherein the vulnerability determination program includes means for generating a vulnerability definition comprising a logical combination of hash values representing resources associated with a vulnerability, and means for comparing the vulnerability definition with the set of hash values representing the resources of the data processing apparatus.

28 The computer program product of claim 26, further comprising:

program code for retrieving vulnerability-resolution instructions relevant to the vulnerability, from a second data processing apparatus.

29. The computer program product of claim 26, further comprising:

program code for retrieving patching code relevant to the vulnerability, from the second data processing system.